

**This Page Is Inserted by IFW Operations  
and is not a part of the Official Record**

## **BEST AVAILABLE IMAGES**

**Defective images within this document are accurate representations of the original documents submitted by the applicant.**

**Defects in the images may include (but are not limited to):**

- **BLACK BORDERS**
- **TEXT CUT OFF AT TOP, BOTTOM OR SIDES**
- **FADED TEXT**
- **ILLEGIBLE TEXT**
- **SKEWED/SLANTED IMAGES**
- **COLORED PHOTOS**
- **BLACK OR VERY BLACK AND WHITE DARK PHOTOS**
- **GRAY SCALE DOCUMENTS**

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

**THIS PAGE BLANK (USPTO)**



JC19 Rec'd PCT/PTO 13 AUG 2001

5000

A34394 PCT USA - 066340.0131

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Matthias Weiss  
Serial No. : 09/868,797  
Filed : June 21, 2001  
For : DEVICE AND METHOD FOR GENERATING AND  
EXECUTING COMPRESSED PROGRAM OF A VERY  
LONG INSTRUCTION WORD PROCESSOR

**CLAIM FOR PRIORITY UNDER 35 U.S.C. § 119**

I hereby certify that this paper is being deposited with the United States  
Postal Service as first class mail in an envelope addressed to:  
Assistant Commissioner for Patents, Washington, D.C. 20231, on:

August 6, 2001

Date of Deposit

Ronald B. Hildreth

Attorney Name

19,498

PTO Reg. No.

August 6, 2001

Date of Signature

Assistant Commissioner for Patents

Washington, D.C. 20231

Sir:

A claim for priority is hereby made under the provisions of 35 U.S.C. § 119 for the  
above-identified PCT application based upon German application 198 59 389.9 filed December  
22, 1998, and International Application PCT/DE99/04050 filed December 21, 1999.

Respectfully submitted

Ronald B. Hildreth

Patent Office Reg. No. 19,498

(212) 408-2544

Attorney for Applicants

Baker Botts L.L.P.  
30 Rockefeller Plaza  
New York NY 10112

**THIS PAGE BLANK (USPTO)**

**PCT INITIAL PROCESSING**

**AUG 17 2001**

**RECEIVED**

**BUNDESREPUBLIK DEUTSCHLAND**

**PRIORITY  
DOCUMENT**  
SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH RULE 17.1(a) OR (b)



REC'D 02 MAR 2000	
WIPO	PCT

**Bescheinigung**

de 99 / 4050

4

Die Anmelderin Technische Universität Dresden in Dresden/Deutschland hat eine Patentanmeldung unter der Bezeichnung

"Verfahren zur Ansteuerung von Funktionseinheiten in einem Prozessor und Prozessoranordnung zur Durchführung des Verfahrens"

am 22. Dezember 1998 beim Deutschen Patent- und Markenamt eingereicht.

Die Anmeldung ist auf die Systemonic AG in Dresden/Deutschland umgeschrieben worden.

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

Die Anmeldung hat im Deutschen Patent- und Markenamt vorläufig das Symbol G 06 F 9/28 der Internationalen Patentklassifikation erhalten.

München, den 14. Februar 2000

**Deutsches Patent- und Markenamt**

**Der Präsident**

Im Auftrag

Hoiß

Aktenzeichen: 198 59 389.9

**LIPPERT, STACHOW, SCHMIDT & PARTNER**

Patentanwälte · European Patent Attorneys · European Trademark Attorneys

P.O.Box 192438 · D-01282 Dresden

Telefon +49 (0) 351.31818-0

Telefax +49 (0) 351.3181833

Ad/ad

22. Dezember 1998

5

**Technische Universität Dresden****01062 Dresden**

10

**Verfahren zur Ansteuerung von Funktionseinheiten in einem  
Prozessor und Prozessoranordnung zur Durchführung des Ver-  
fahrens**

15

**Zusammenfassung**

20

25

30

35

Der Erfindung, die ein Verfahren zur Ansteuerung von Funktionseinheiten in einem Prozessor, bei dem in einer Konfiguration eine aus einer Übersetzung eines Programmcodes stammende Folge von Primärinstruktionsworten aus mehreren Instruktionsteilen komprimiert und als Folge von zugehörigen Programmworten gespeichert wird, und eine Prozessoranordnung zur Durchführung des Verfahrens betrifft, liegt die Aufgabe zugrunde, unter Beibehaltung einer geringen Programmwortbreite die Arbeitsgeschwindigkeit anwendungsspezifisch zu erhöhen. Verfahrensseitig wird dies dadurch gelöst, daß ein Programmwort ein erstes Kennzeichen eines Primärinstruktionswortes und Instruktionsteile, die das zu dem Programmwort gehörige Primärinstruktionswort von dem zu dem Kennzeichen gehörigen Primärinstruktionswort unterscheidenden, enthält. Mittels dem in dem Programmwort enthaltenen ersten Kennzeichen wird ein Sekundärinstruktionswort erzeugt, indem aus einem vorhergehenden Sekundärinstruktionswort die in dem Programmwort enthaltenen Instruktionsteile ausgetauscht werden. Anordnungsseitig wird die Aufgabe dadurch gelöst, daß der Instruktionsteilpuffer aus einem Speicher mit wahlweisem zeilenweisen Zugriff besteht. (Fig. 1)

22.12.98

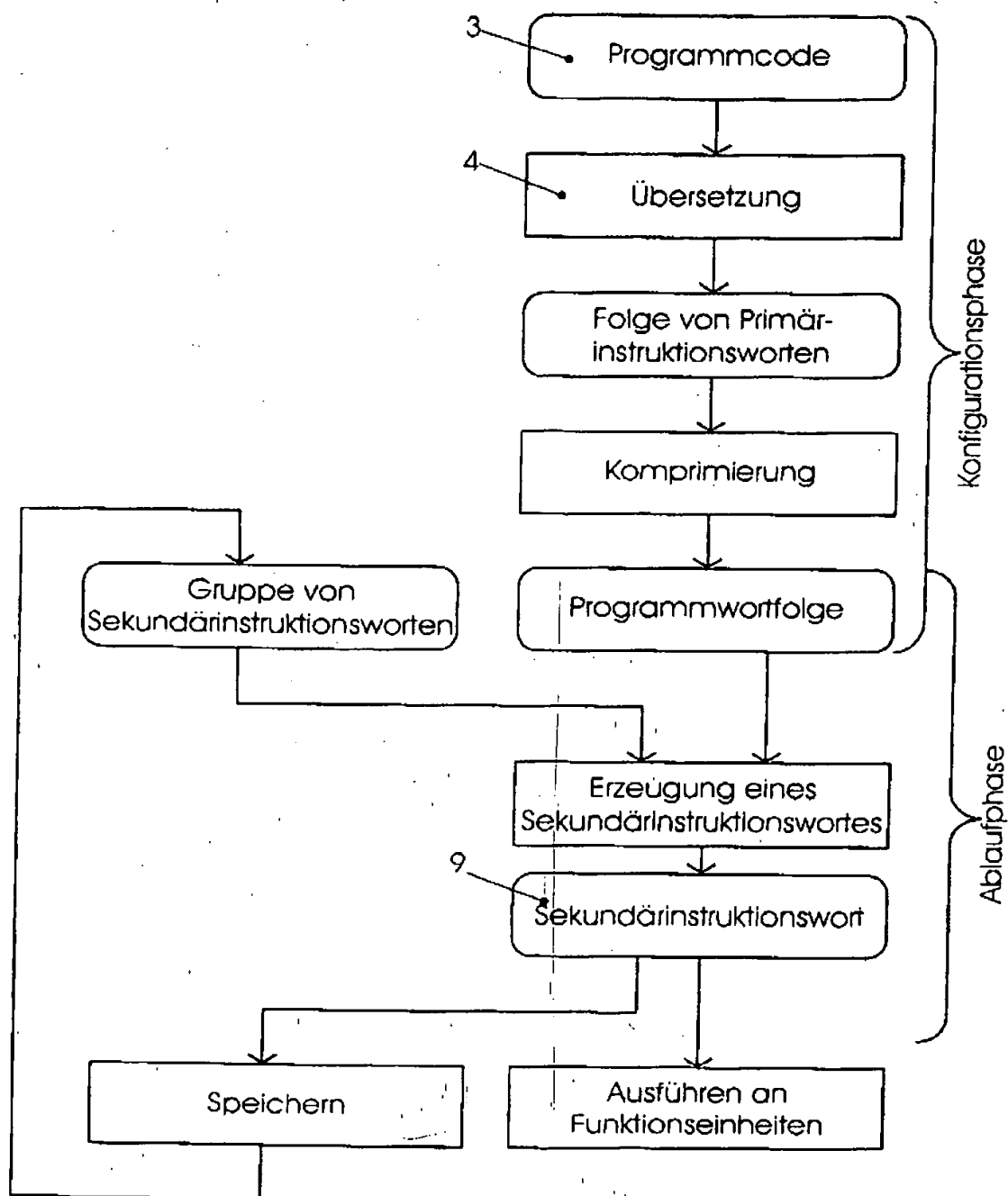


Fig. 1

22.12.98

1

**LIPPERT, STACHOW, SCHMIDT & PARTNER**  
Patentanwälte · European Patent Attorneys · European Trademark Attorneys  
P.O. Box 19 24 38 · D-01282 Dresden  
Telefon +49 (0) 3 51 3 18 18-0  
Telefax +49 (0) 3 51 3 19 18 33

Ad/ad

22. Dezember 1998

5

**Technische Universität Dresden**  
**01062 Dresden**

10

**Verfahren zur Ansteuerung von Funktionseinheiten in einem  
Prozessor und Prozessoranordnung zur Durchführung des Ver-  
fahrens**

15

20

Die Erfindung betrifft ein Verfahren zur Ansteuerung von Funktionseinheiten in einem Prozessor, bei dem in einer Konfiguration eine aus einer Übersetzung eines Programmcodes stammende Folge von Primärinstruktionsworten aus mehreren Instruktionsteilen komprimiert und als Folge von zugehörigen Programmworten gespeichert wird. Dabei werden in einer nachfolgenden Ausführungsphase aus der Folge von Programmworten aufeinanderfolgende Sekundärinstruktionsworte aus mehreren Instruktionsteilen in der zur Ansteuerung aller Funktionseinheiten benötigten vollen Instruktionsteilbreite erzeugt.

25

30

Die Erfindung betrifft auch eine Prozessoranordnung zur Durchführung des Verfahrens mit Funktionseinheiten, einem diesen Funktionseinheiten zugeordneten Instruktionsteilwortspeicher und einem Instruktionsteilwortpuffer zur Speicherung bereits erzeugter Instruktionsteilworte mit einer Breite, die mindestens gleich der Bitbreite des Sekundärinstruktionswortes ist.

35

Prozessoranordnungen der eingangs genannten Art beinhalten Funktionseinheiten, die parallel zueinander arbeiten und die zu jedem Takt von einem Instruktionsteilwort angesteuert werden. Das jeweilige Instruktionsteilwort wird aus einem Programmwort gewonnen, das einem Programmspeicher entnommen wird.



22.12.98

2

Die Instruktionsworte ihrerseits bestehen aus mehreren Instruktionswortteilen, wobei je ein Instruktionswortteil der Ansteuerung je einer Funktionseinheit dient.

5 Zur Erhöhung der Leistungsfähigkeit von Prozessoranordnungen besteht das Bestreben nach einer Vergrößerung der Verarbeitungsbreite, was eine Erhöhung der Anzahl von Funktionseinheiten erforderlich werden läßt. Damit vergrößert sich im  
10 allgemeinen die Bitbreite der Instruktionsworte und damit auch der Programmworte. Dies hat eine Bereitstellung eines entsprechenden Speicherplatzes im Programmspeicher zur Folge, der der wesentliche Flächenteile auf dem Halbleiterchip einnimmt.

15 Da die Größe der Programmworte die Größe des Programmspeichers bestimmen, ist es das Bestreben, die Größe der Programmworte zu verringern, um eine Verringerung des Speicherplatzbedarfes zu erreichen. Hierfür sind mehrere Komprimierungsverfahren bekannt. Das naheliegendste Verfahren ist in dem Bericht von  
20 Weiss H., Fettweis, G. "Dynamic Codewidth Reduction for VLIW Instruction Set Architectures in Digital Signal Processors" (Tagungsbericht des 3. Int. Workshop on Signal an Image Processing IWSIP '96 S. 517 bis 520) beschrieben. Darin werden  
25 die Programmworte aus hintereinanderliegenden Primärinstruktionsworten so zusammengestellt, daß anschließend daraus wieder Sekundärinstruktionsworte erstellt werden können, indem ein einmal erzeugtes Sekundärinstruktionswort (VLIW) in einen Instruktionswortspeicher geschrieben wird und zur Erzeugung  
30 des nächsten Sekundärinstruktionswortes bei dem gespeicherten Sekundärinstruktionswort nur die Instruktionswortteile ausgetauscht werden, die sich von dem gespeicherten Sekundärinstruktionswort zu dem zu erzeugenden Sekundärinstruktionswort unterscheiden. Mithin braucht in dem Programmwort nur die Information enthalten zu sein, welches Instruktionswortteil sich unterscheidet und mit welchem Inhalt es sich unterscheidet.  
35 Damit ist es möglich, die Programmworte sehr schmal zu gestalten und damit Speicherplatz zu sparen.

Bei größeren Unterschieden zwischen dem gespeicherten und dem

herzustellenden Sekundärinstruktionswort ist allerdings die Breite des Programmwortes zu vergrößern, wenn diese größeren Unterschiede häufig auftreten, was den Nachteil eines größeren Speicherplatzes mit sich bringt, oder die Unterschiede sind  
5 über mehrere Programmwörter zu verteilen. Damit ist das Sekundärinstruktionswort aus mehreren Programmworten über mehrere Takte herzustellen. Daraus entsteht der Nachteil eines größeren Zeitaufwandes.

10 Der Erfindung liegt damit die Aufgabe zugrunde, unter Beibehaltung einer geringen Programmwortbreite die Arbeitsgeschwindigkeit anwendungsspezifisch zu erhöhen.

Verfahrensseitig wird die Aufgabe dadurch gelöst, daß im Ergebnis der Konfiguration ein Programmwort ein erstes Kennzeichen eines Primärinstruktionswortes aus einer ersten Gruppe vorhergehender Primärinstruktionsworte, welches mit dem zum Programmwort zugehörigen Primärinstruktionswort die größte Übereinstimmung aufweist, und Instruktionswortteile, die das  
15 zu dem Programmwort gehörige Primärinstruktionswort von dem zu dem Kennzeichen gehörigen Primärinstruktionswort unterscheidenden, enthält. In der Ausführungsphase wird eine der ersten Gruppe in der Anzahl entsprechende zweite Gruppe von Sekundärinstruktionsworten mit je einer zweiten Kennzeichnung  
20 versehen gespeichert. Mittels dem in dem Programmwort enthaltenen ersten Kennzeichen wird ein dem zugehörigen Primärinstruktionswort entsprechendes Sekundärinstruktionswort aus der zweiten Gruppe über das zugehörige zweite Kennzeichen ermittelt wird und das dem Programmwort entsprechende Sekundärinstruktionswort erzeugt, indem in dem Sekundärinstruktionswort aus der zweiten Gruppe die in dem Programmwort enthaltenen Instruktionswortteile ausgetauscht werden.  
25  
30

Bereits bei der Konfigurationsphase können die Programmworte  
35 so aufgebaut werden, daß sie nur die geringstmögliche Information zur späteren Erzeugung eines Sekundärinstruktionswortes in der Ausführungsphase beinhalten. Dies wird damit erreicht, daß auf die vorhergehenden Primärinstruktionswörter zurück-

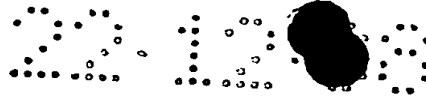
22.12.98

4

gegriffen wird, die die größte Übereinstimmung mit dem zu konfigurierenden Primärinstruktionswort aufweisen. Da in der Ausführungsphase die Sekundärinstruktionswörter in der gleichen Breite und in der gleichen Reihenfolge wie die Primärinstruktionswörter erzeugt werden sollen, wird in der Ausführungsphase praktisch in umgekehrter Reihenfolge zur der Konfigurationsphase verfahren und jeweils ein bereits erzeugtes Sekundärinstruktionswort, welches dem Primärinstruktionswort entspricht, das in der Konfigurationsphase einem aktuellen Primärinstruktionswort am ähnlichsten ist, zur Erzeugung eines neuen Sekundärinstruktionswortes verwendet wird. Da stets auf das Instruktionswort mit der größten Ähnlichkeit oder der größten Übereinstimmung zurückgegriffen wird, läßt sich die Menge der zur Erzeugung eines neuen Sekundärinstruktionswortes erforderliche Information so klein wie möglich halten. Damit kann der Speicherbedarf eines Programmspeichers minimiert werden.

In einer Ausführungsform des Verfahrens ist vorgesehen, daß die erste Gruppe aus einer ersten Anzahl der dem jeweiligen Primärinstruktionswort direkt vorausgehenden Primärinstruktionswörtern besteht. Die zweite Gruppe besteht aus einer zweiten Anzahl von Sekundärinstruktionswörtern, die mindestens gleich der ersten Anzahl ist, wobei jedes Sekundärinstruktionswort vor der Erzeugung des nächstfolgenden Sekundärinstruktionswortes der Gruppe als letztes hinzugefügt und das über die zweite Anzahl hinausgehende zuerst hinzugefügte Sekundärinstruktionswort aus der zweiten Gruppe entfernt wird. Damit besteht die Gruppe stets aus den unmittelbar vorhergehenden Instruktionswörtern, wovon eines eine größtmögliche Ähnlichkeit aufweist, wenn sich die Instruktionswörter innerhalb einer Gruppe inhaltlich unterscheiden.

Eine andere Möglichkeit der Ausgestaltung des Verfahrens besteht darin, daß das neu erzeugte Sekundärinstruktionswort der zweiten Gruppe hinzugefügt wird, indem dieses anstelle des seiner Erzeugung dienenden Sekundärinstruktionswortes gespeichert wird.



Mit dieser Variante des Verfahrens wird es ausgeschlossen, daß sich die Gruppe mit inhaltsgleichen Instruktionswörtern beim mehrmaligen Durchlaufen ein und derselben Funktion anfüllt.

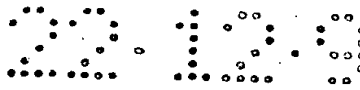
5 Eine andere Möglichkeit besteht darin, daß das neu erzeugte Sekundärinstruktionswort nicht gespeichert wird. Dabei wird bei jeder Erzeugung eines Sekundärinstruktionswortes stets auf den gleichen Vorrat von einmal erzeugten Instruktionswörtern zurückgegriffen, was günstig ist, wenn diese gespeicherten Instruktionswörter als Musterwörter für die Erzeugung anderer  
10 Sekundärinstruktionswörter geeignet sind, dabei hierbei Speichervorgänge vermieden werden können.

15 In einer Ausgestaltung des Verfahrens ist vorgesehen, daß das erste Kennzeichen als Abstand des zum jeweiligen Programmwort gehörigen Primärinstruktionswortes zu dem Primärinstruktionswort mit der größten Übereinstimmung gebildet wird.

20 Eine weitere Ausgestaltung des Verfahrens sieht vor, daß das zweite Kennzeichen aus einer dem ersten Kennzeichen entsprechenden Adresse eines vorhergehenden Sekundärinstruktionswortes in einem der Abspeicherung der zweiten Gruppe dienenden Speicher besteht.

25 Diese beiden Ausgestaltungen zeigen eine besonders einfache Erzeugung von Merkmalen.

30 Eine Reduzierung der Programmwortbreite und damit verbunden eine Verringerung der Breite des Programmspeichers kann dadurch erreicht werden, daß das Programmwort aus einer Anzahl von Instruktionswortteilen besteht, die der innerhalb der Konfiguration am häufigsten auftretenden Anzahl zu unterscheidender Instruktionswortteile entspricht und daß zur Zusammensetzung von Sekundärinstruktionsworten, die zu dem der Erzeugung dienenden Sekundärinstruktionswort mehr als die in einem  
35 Programmwort abgelegte Anzahl von Instruktionsworten benötigt, mehrere Programmwörter verwendet werden. Somit genügt die Breite des Programmspeichers den häufigsten Anwendungsfällen,



wobei auch keine Verzögerung in der Erstellung des Sekundärinstruktionswortes eintritt. Lediglich in den relativ wenigen Fällen, in denen die Anzahl der zu ändernden Instruktionswortteile die Breite eines Programmwortes überschreitet, werden zwei oder mehr Programmworte in zwei oder mehr Takten zur Herstellung des Sekundärinstruktionswortes benötigt.

Eine weitere Verringerung der Programmwortbreite kann dadurch erreicht werden, daß die Instruktionswortteile in einem Programmwort komprimiert werden. Dies geschieht dadurch, daß die Bitbreite in dem Maße verringert wird, wie eine Darstellung der am häufigsten vorkommenden Instruktionswortteile möglich ist, und daß bei einem Auftreten von Instruktionswortteilen, die zur Darstellung eine größere Bitbreite benötigen, mehrere Programmwörter verwendet werden.

Für eine derartige Komprimierung des Programmwortes ist es insbesondere zweckmäßig, daß die Breite der Instruktionswortteile in dem Programmwort halbiert wird und zur Darstellung der Instruktionswortteile ein oder zwei Programmwörter vorgesehen sind. Hat beispielsweise normalerweise das Programmwort eine Breite von 8 Bit, so stehen 256 Kombinationen zur Verfügung. Ein großer Teil dieser Kombinationen wird jedoch nur äußerst selten oder gar nicht benötigt. Dementsprechend kann das Programmwort mit einer Breite von 4 Bit vorgesehen werden, was für die 16 häufigsten Kombinationen zur Darstellung ausreichend ist. Soll eine andere als eine dieser Kombination dargestellt werden, werden hierfür zwei oder mehr Programmwörter verwendet. Da dies jedoch nur sehr selten auftritt, ist die erreichte Einsparung an Speicherplatz für den Programmspeicher größer als der mögliche Zeitaufwand zur Erzeugung seltener Kombinationen.

Die Aufgabenstellung wird erfindungsgemäß auch durch eine Prozessoranordnung gelöst, die dadurch gekennzeichnet ist, daß der Instruktionswortpuffer aus einem Speicher mit wahlfreiem zeilenweisen Zugriff besteht. Im Unterschied zu dem dargestellten Stand der Technik besteht damit die Möglichkeit,

22.12.98

7

direkt auf verschiedene abgespeicherte Instruktionsworte zur Herstellung eines neuen Sekundärinstruktionswortes zurückzugreifen. Damit muß nicht irgendein gespeichertes Instruktionswort der Erzeugung eines neuen Sekundärinstruktionswortes dienen, sondern es kann dasjenige abgespeicherte Instruktionswort verwendet werden, das mit dem zu erzeugenden Sekundärinstruktionswort die größte Übereinstimmung aufweist. Damit ist der Änderungsaufwand relativ klein, was nur eine geringe Bitbreite des Programmwortes und damit einen kleinen Programmspeicher erfordert.

Die Erfindung soll nachfolgend anhand eines Ausführungsbeispiels näher erläutert werden. In den zugehörigen Zeichnungen zeigt

Fig. 1 ein Ablaufschema des erfindungsgemäßen Verfahrens und

Fig. 2 das Prinzip der erfindungsgemäßen Erzeugung der Primär- und Sekundärinstruktionsworte

Bei dem in Fig. 1 dargestellten Verfahren zur Ansteuerung von Funktionseinheiten 1 in einem Prozessor 2 wird aus einem Programmcode 3 mittels einer Übersetzung 4 in einer Konfigurationsphase eine Folge von Primärinstruktionsworten 5 aus mehreren Instruktionswortteilen 6 erzeugt. Diese Folge von Primärinstruktionsworten 5 wird komprimiert und als Folge von zugehörigen Programmworten 7 in einem Programmspeicher 8 gespeichert.

Die Programmworte 7 bestehen aus einer Anzahl von Instruktionswortteilen 6, die der innerhalb der Konfiguration am häufigsten auftretenden Anzahl zu unterscheidender Instruktionswortteile 6 entspricht. Zur Zusammensetzung von Sekundärinstruktionsworten 9 können mehr als die in einem Programmwort 7 abgelegte Anzahl von Instruktionswortteilen 6 benötigt werden. Dazu werden dann mehrere Programmwörter 7 verwendet.

In einer nachfolgenden Ausführungsphase werden aus der Folge



8

von Programmworten 7 aufeinanderfolgende Sekundärinstruktionsworte 9 aus mehreren Instruktionsteilen 6 in der zur Ansteuerung aller Funktionseinheiten 1 benötigten vollen Instruktionsteilbreite erzeugt.



5

Im Ergebnis der Konfiguration weist ein Programmwort 7 ein erstes Kennzeichen 10 eines Primärinstruktionswortes 5 aus einer ersten Gruppe 11 vorhergehender Primärinstruktionsworte 5, welches mit dem zum Programmwort 7 zugehörigen Primärinstruktionswort 5 die größte Übereinstimmung aufweist, und Instruktionsteile 6, die das zu dem Programmwort 7 gehörige Primärinstruktionswort 5 von dem zu dem Kennzeichen gehörigen Primärinstruktionswort 5 unterscheidenden, enthält. Das erste Kennzeichen 10 wird als Abstand des zum jeweiligen Programmwort 7 gehörigen Primärinstruktionswortes 5 zu dem Primärinstruktionswort 5 mit der größten Übereinstimmung gebildet.

In der Ausführungsphase wird eine der ersten Gruppe 11 in der Anzahl entsprechende zweite Gruppe 12 von Sekundärinstruktionswörtern 9 mit je einem zweiten Kennzeichen 13 versehen gespeichert. Das zweite Kennzeichen 13 wird aus einer Zeilennummer eines der Abspeicherung der zweiten Gruppe 12 dienenden Speichers 14 gebildet.

Mittels dem in dem Programmwort 7 enthaltenen ersten Kennzeichen 10 wird ein dem zugehörigen Primärinstruktionswort 5 entsprechendes Sekundärinstruktionswort 9 aus der zweiten Gruppe 12 über das zugehörige zweite Kennzeichen 13 ermittelt. Das dem Programmwort entsprechende Sekundärinstruktionswort 9 wird erzeugt, indem in dem Sekundärinstruktionswort 9 aus der zweiten Gruppe 12 die in dem Programmwort 7 enthaltenen Instruktionsteile 6 ausgetauscht werden.

Die erste Gruppe 11 besteht aus einer ersten Anzahl der dem jeweiligen Primärinstruktionswort 5 vorausgehenden Primärinstruktionswörtern 5. Die zweite Gruppe 12 besteht aus einer zweiten Anzahl von Sekundärinstruktionswörtern 9, die mindestens gleich der ersten Anzahl ist, wobei jedes Sekundärin-



22.12.98

9

struktionswort 9 vor der Erzeugung des nächstfolgenden Sekundärinstruktionswortes 9 der zweiten Gruppe 12 als letztes hinzugefügt werden kann. Das über die zweite Anzahl hinausgehende zuerst hinzugefügte Sekundärinstruktionswort 9 wird aus der zweiten Gruppe 12 entfernt.

5



201000

10

**LIPPERT, STACHOW, SCHMIDT & PARTNER**

Patentanwälte · European Patent Attorneys · European Trademark Attorneys

P.O.Box 19 24 38 · D-01282 Dresden

Telefon +49 (0) 3 51 3 18 16-0

Telefax +49 (0) 3 51 3 18 19 33

Ad/ad

22. Dezember 1998

5

Technische Universität Dresden

01062 Dresden

10

Verfahren zur Ansteuerung von Funktionseinheiten in einem  
Prozessor und Prozessoranordnung zur Durchführung des Ver-  
fahrens

15

Bezugszeichenliste

- |    |                            |
|----|----------------------------|
| 1  | Funktionseinheit           |
| 2  | Prozessor                  |
| 20 | 3 Programmcode             |
|    | 4 Übersetzung              |
|    | 5 Primärinstruktionswort   |
|    | 6 Instruktionswortteil     |
|    | 7 Programmwort             |
| 25 | 8 Programmspeicher         |
|    | 9 Sekundärinstruktionswort |
|    | 10 erstes Kennzeichen      |
|    | 11 erste Gruppe            |
|    | 12 zweite Gruppe           |
| 30 | 13 zweites Kennzeichen     |
|    | 14 Speicher                |



11

**LIPPERT, STACHOW, SCHMIDT & PARTNER**  
Patentanwälte · European Patent Attorneys · European Trademark Attorneys  
P.O.Box 19 24 38 · D-01282 Dresden  
Telefon +49 (0) 3 51 3 18 18-0  
Telefax +49 (0) 3 51 3 18 18 33

Ad/ad

22. Dezember 1998

5

Technische Universität Dresden  
01062 Dresden

10

**Verfahren zur Ansteuerung von Funktionseinheiten in einem  
Prozessor und Prozessoranordnung zur Durchführung des Ver-  
fahrens**

15

**Patentansprüche**

20

25

30

35

1. Verfahren zur Ansteuerung von Funktionseinheiten in einem Prozessor, bei dem in einer Konfiguration eine aus einer Übersetzung eines Programmcodes stammende Folge von Primärinstruktionsworten aus mehreren Instruktionswortteilen komprimiert und als Folge von zugehörigen Programmworten gespeichert wird und bei dem in einer nachfolgenden Ausführungsphase aus der Folge von Programmworten aufeinanderfolgende Sekundärinstruktionsworte aus mehreren Instruktionswortteilen in der zur Ansteuerung aller Funktionseinheiten benötigten vollen Instruktionswortbreite erzeugt werden, d a d u r c h g e k e n n z e i c h n e t,

daß im Ergebnis der Konfiguration ein Programmwort (7) ein erstes Kennzeichen (10) eines Primärinstruktionswortes (5) aus einer ersten Gruppe (11) vorhergehender Primärinstruktionsworte (5), welches mit dem zum Programmwort (7) zugehörigen Primärinstruktionswort (5) die größte Übereinstimmung aufweist, und Instruktionswortteile (6), die das zu dem Programmwort (7) gehörige Primärinstruktionswort (5) von dem zu dem ersten Kennzeichen (10) gehörigen Primärinstruktionswort (5) unterscheidenden, enthält,



12

daß in der Ausführungsphase eine der ersten Gruppe (11) in der Anzahl entsprechende zweite Gruppe (12) von Sekundärinstruktionsworten (9) mit je einem zweiten Kennzeichen (13) versehen gespeichert wird, und

5

daß mittels dem in dem Programmwort (7) enthaltenen ersten Kennzeichen (10) ein dem zugehörigen Primärinstruktionswort (5) entsprechendes Sekundärinstruktionswort (9) aus der zweiten Gruppe (12) über das zugehörige zweite Kennzeichen (13) ermittelt wird und das dem Programmwort (7) entsprechende Sekundärinstruktionswort (9) erzeugt wird, indem in dem Sekundärinstruktionswort (9) aus der zweiten Gruppe (12) die in dem Programmwort (7) enthaltenen Instruktionswortteile (6) ausgetauscht werden.

10

15

2. Verfahren nach Anspruch 1, d a d u r c h g e k e n n - z e i c h n e t, daß die erste Gruppe (11) aus einer ersten Anzahl der dem jeweiligen Primärinstruktionswort (5) direkt vorausgehenden Primärinstruktionswörtern (5) besteht und

20

daß die zweite Gruppe (12) aus einer zweiten Anzahl von Sekundärinstruktionswörtern (9) besteht, die mindestens gleich der ersten Anzahl ist, wobei jedes Sekundärinstruktionswort (9) vor der Erzeugung des nächstfolgenden Sekundärinstruktionswortes (9) der zweiten Gruppe (12) als letztes hinzugefügt und das über die zweite Anzahl hinausgehende zuerst hinzugefügte Sekundärinstruktionswort (9) aus der zweiten Gruppe (12) entfernt wird.

25

30

3. Verfahren nach Anspruch 1, d a d u r c h g e k e n n - z e i c h n e t, daß das neu erzeugte Sekundärinstruktionswort (9) der zweiten Gruppe (12) hinzugefügt wird, indem dieses anstelle des seiner Erzeugung dienenden Sekundärinstruktionswortes (9) gespeichert wird.

35

4. Verfahren nach Anspruch 1, d a d u r c h g e k e n n - z e i c h n e t, daß das neu erzeugte Sekundärinstruk-



13

tionswort (9) nicht gespeichert wird.

5. Verfahren nach einem der Ansprüche 1 bis 4, d a d u r c h  
g e k e n n z e i c h n e t, daß das erste Kennzeichen  
(10) als Abstand des zum jeweiligen Programmwort gehörigen  
Primärinstruktionswortes zu dem Primärinstruktionswort mit  
der größten Übereinstimmung gebildet wird.
6. Verfahren nach einem der Ansprüche 1 bis 5, d a d u r c h  
g e k e n n z e i c h n e t, daß das zweite Kennzeichen  
(13) aus einer dem ersten Kennzeichen entsprechenden  
Adresse eines vorhergehenden Sekundärinstruktionswortes  
in einem der Abspeicherung der zweiten Gruppe (12) dienen-  
den Speichers (14) besteht.
7. Verfahren nach einem der Ansprüche 1 bis 6, d a d u r c h  
g e k e n n z e i c h n e t, daß das Programmwort (7) aus  
einer Anzahl von Instruktionsteilen (6) besteht, die  
der innerhalb der Konfiguration am häufigsten auftretenden  
Anzahl zu unterscheidender Instruktionsteile (6)  
entspricht und daß zur Zusammensetzung von Sekundärin-  
struktionsteilen (9), die zu dem der Erzeugung dienenden  
Sekundärinstruktionsteil (9) mehr als die in einem Pro-  
grammwort (7) abgelegte Anzahl von Instruktionsteilen (9)  
benötigt, mehrere Programmwörter (7) verwendet werden.
8. Verfahren nach einem der Ansprüche 1 bis 7, d a d u r c h  
g e k e n n z e i c h n e t, daß die Instruktionsteile  
(6) in einem Programmwort (7) komprimiert werden, indem  
die Bitbreite in dem Maße verringert wird, wie eine Dar-  
stellung der am häufigsten vorkommenden Instruktionsteile  
(6) möglich ist, und daß bei einem Auftreten von  
Instruktionsteilen (6), die zur Darstellung eine grö-  
ßere Bitbreite benötigen, mehrere Programmwörter (7) ver-  
wendet werden.
9. Verfahren nach Anspruch 8, d a d u r c h g e k e n n -  
z e i c h n e t, daß die Breite der Instruktionsteile

22.12.98

14

(6) in dem Programmwort (7) halbiert wird und zur Darstellung der Instruktionswortteile (6) ein oder zwei Programmwörter (7) vorgesehen sind.

- 5      10. Prozessoranordnung zur Durchführung des Verfahrens mit Funktionseinheiten, einem diesen Funktionseinheiten zugeordneten Instruktionswortspeicher und einem Instruktionswortpuffer zur Speicherung bereits erzeugter Instruktionswörter mit einer Breite, die mindestens gleich der
- 10      Bitbreite des Sekundärinstruktionswortes ist, dadurch gekennzeichnet, daß der Instruktionswortpuffer aus einem Speicher (14) mit wahlweisem zeilenweisen Zugriff besteht.
- 15

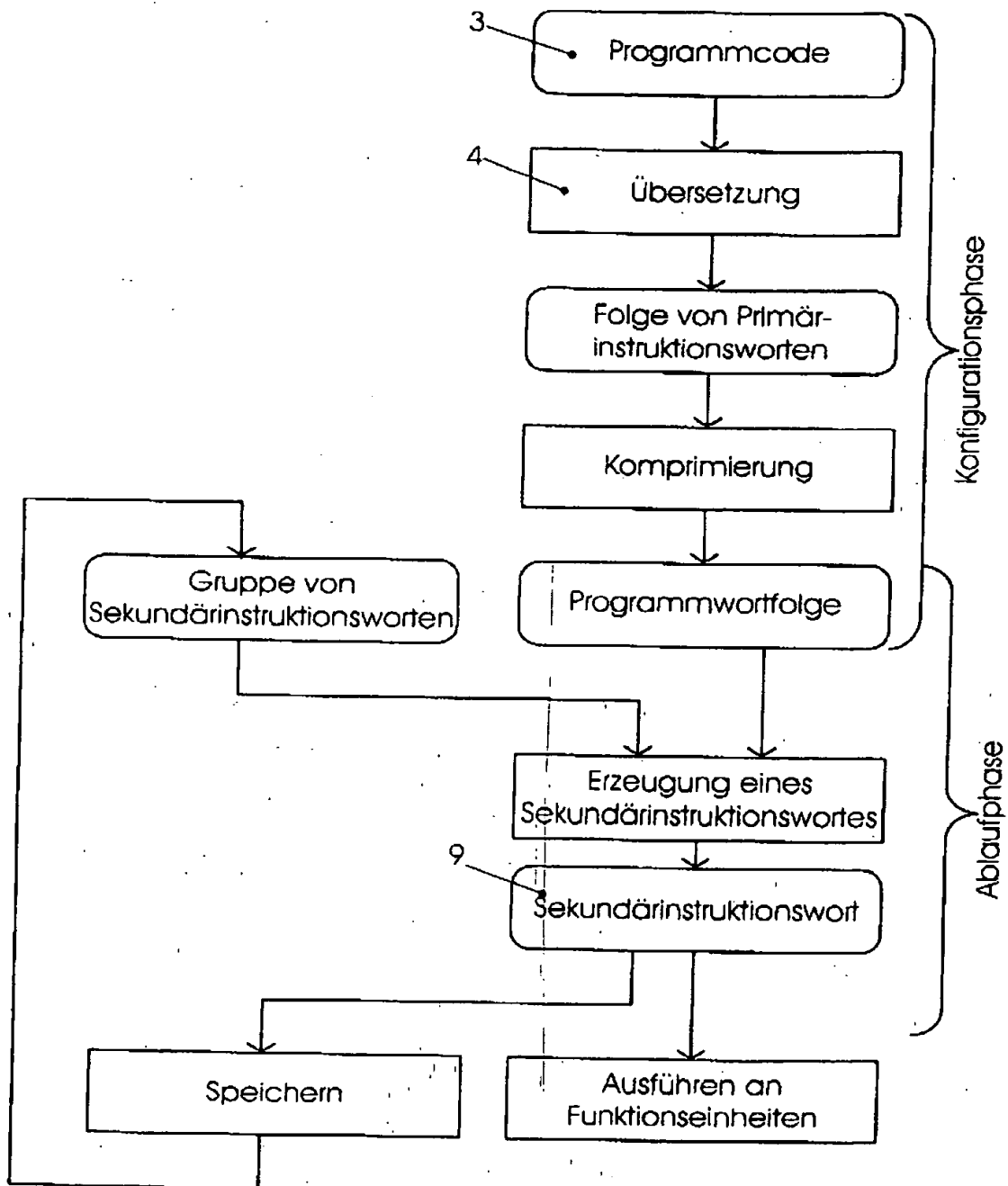


Fig. 1

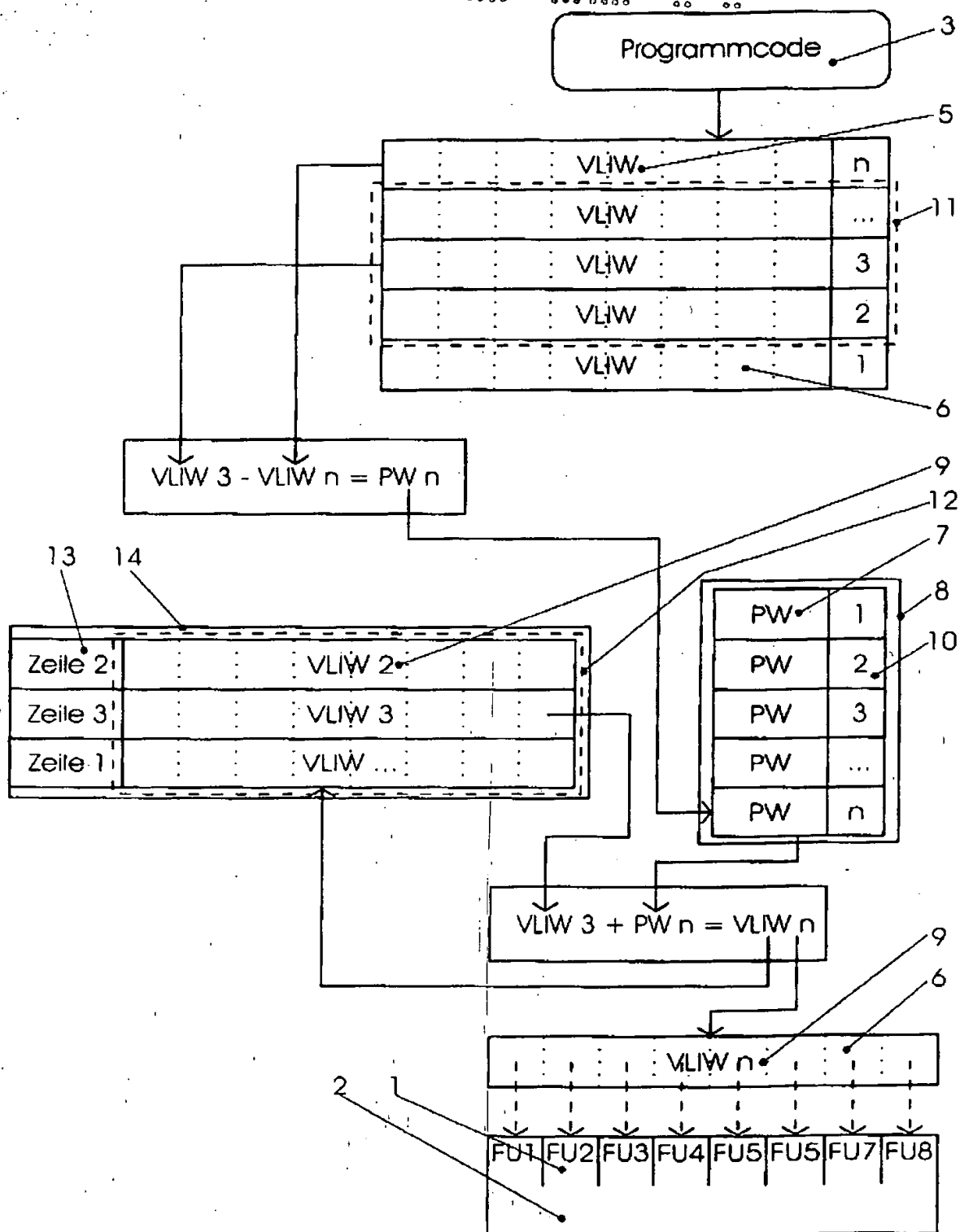


Fig. 2

**LIPPERT, STACHOW, SCHMIDT & PARTNER**

Patentanwälte · European Patent Attorneys · European Trademark Attorneys

## Gebührenblatt

zur deutschen Patentanmeldung:

**Titel:** Verfahren zur Ansteuerung von Funktionseinheiten in einem Prozessor und Prozessoranordnung zur Durchführung des Verfahrens

**Anmelder:** Technische Universität Dresden

**Unser Zeichen:** D680087DE

**Gebührenmarken:**

